Node Breaker Model Based Transient Stability Simulations including Protective Devices Modeling and Time Coordination

Vibhuti Sahu, Gurunath Gurrala, Sanjeev Bangali and Sarasij Das
Department of Electrical Engineering
Indian Institute of Science, Bangalore, India

***Abstract*—Modeling protection devices in the Transient Stability Assessment (TSA) of power systems results in an accurate estimation of system stability and behaviour. Miscoordination/misoperations of the relays and circuit breakers (CBs) result in higher order contingencies, $> N-1$. Substation (SS) configurations and associated CB operations are usually ignored in TSA, even though the relays are modeled in detail. Replicating such scenarios needs manual fabrications in Bus-Branch (BB) models used in commercial TSA tools, which can be eliminated using Node-Breaker (NB) models. Also, incorporating relay algorithms, including the realistic CB operations and coordinating their timings, increases the complexity of the TSA code for large systems. The development of TSA tools to simulate the practical operation of CBs due to protective relay decisions similar to the physical substation operation is essential. This paper attempts in this direction. This paper uses a systematic approach to convert BB models to NB models. The Sparse Tableau Approach (STA) is used to represent the system in the NB model. It provides an algorithmic approach for automatically placing differential, distance and under/over frequency relays. It also gives programming logic to coordinate the timings between relays and the associated CBs. The program complexity increases with the size of the system as the number of CBs and the associated timers significantly increase based on the station configuration. The simulation results demonstrate the scalability and effectiveness of the proposed framework for the WECC 9 bus, New England 39 bus, and Polish 2383 bus systems. The WECC 9 bus system results are compared and validated with the commercial software PSS®E.**

***Index Terms*—Sparse Tableau Approach, Node-Breaker, Transient Stability Simulation, Relay Time Coordination**

I. List of Abbreviations

N_{st}	Number of stations
N_{Se}	Total number of series elements
N_{Sh}	Total number of shunt elements
N_{So}	Total number of source elements
$N_{\mathrm{st}}^{\mathrm{ele}}$	Total number of elements except circuit breakers in a station
$N_{\mathrm{node}}^{\mathrm{ele}}$	Total number of elements connected to a node
N^{line}	Total number of lines in the system
N^{trafo}	Total number of transformers in the system
$N_{\mathrm{cum}}^{\mathrm{nodes}}(k)$	Total number of cummulative nodes till k^{th} station
$N_{\mathrm{st}}^{\mathrm{phy}}$	Number of physical buses in a station
$N_{\mathrm{st}}^{\mathrm{itmN}}$	Number of intermediate nodes created

This work is carried out under the project "BLACK OUT LOOK-AHEAD SIMULATION TOOL (BLAST)" funded by National Super-computing Mission, a joint mission by Department of Science and Technology (DST) and Department of Electronics and Information Technology (DeitY), India, DST/NSM/R&D/HPC/Applications/2021/03.35.

II. Introduction

Evaluation of the power system after a major disturbance depends on the interaction between the protection system and dynamics governed by generators, loads and control devices [1]. Misoperation of the protection system or incorrect setting can lead to blackouts [2], [3]. In most of the literature, protective actions in dynamic simulations are mimicked by applying the fault for a very short duration, assuming that the protection system clears the fault. This completely neglects the CB operations and their timings, which pose significant challenges in interpreting the sequence of events records when replicating a blackout scenario during postmortem analysis using TSA tools, even though they model relay algorithms [4]. Hence, modeling detailed station configurations, including protection algorithms and associated CB operations in stability simulations, would result in high-fidelity simulations capable of replicating practical operating scenarios.

Relay settings based on NERC standards augmented with generic protection relay models to show the impact of protection devices in dynamic simulations are shown in [2]. In [5], modeling of the protection system in time domain simulation is performed with a method to determine the location of the mis-operating relays at the planning phase. A three-layered model for protective relays is proposed in [6], allowing easy involvement of accurate dynamics. A method to represent protective relay models in stability simulations with inverter based resources (IBRs) is proposed in [7]. This literature proves the importance of modeling relays in transient simulation, albeit modeling all the protection devices in TSA and representing and maintaining protection system data is considered an intractable task [8], [9]. Hence, a systematic method to determine essential relays to be modeled in transient stability studies is presented in [8]. An iterative algorithm that uses (i) apparent impedance monitoring, and (ii) the minimum voltage evaluation (MVE) for identifying critical distance relays in dynamic simulation is proposed in [9]. Moreover, in literature all the relay assessments in dynamic simulations are performed on Bus-Branch (BB) models, which limits the substation level analyses. In [10], various utilities have mentioned the importance of NB models in simulations and their inclusion in various software packages. For example, in a substation, if a CB fails to open after the relay trip command within a specified time (typically CB operating time of 1.5 to 3 cycles), a breaker failure protection (BFP) will act.

Depending on the substation configuration, the BFP can lead to additional elements tripping and delay in clearing the fault. Similarly, a busbar relay clearing a fault in a substation with a breaker and half scheme differs from a main and transfer scheme. In the former case, bus-splitting will happen; in the latter case, all the substation elements will be tripped. To create such realistic scenarios and study their impacts, one must understand the end effect of these scenarios and make suitable modifications to the Y-Bus matrix in the existing commercial TSA tools. This is because these tools do not explicitly model the CBs and their timing, even though they take station configurations as input [10].

In [11]–[14], the importance of considering breaker failures for system reliability assessment has been addressed. Neural networks are employed to model the protection systems in [11], dealing with the uncertainties involved with relay and circuit-breaker operation messages to identify fault sections after contingencies. Two types of protection failures, undesired-tripping mode and fail-to-operate mode, and their impact on reliability modeling are discussed in [12]. In [13], an event-based dual-timer BFP scheme is implemented with an additional signal depending on the distance of the multi-phase fault from the busbars of the power plant for transient stability simulations. A study in [14] proposes a generalized analytical methodology to identify the failure events due to the stuck breaker condition through search algorithms and repeated matrix operations. In [15], the importance of using NB models for steady state cascade analysis has been examined for a practical system to meet regulatory compliance. In [16], NB modeling for optimal power flow using Sparse Tableau Approach (STA) was proposed. In [17], we have proposed NB model for Transient Stability Analysis (TSA) using STA for the first time in the literature. However, the relay and CB timings coordination is not included.

Existing TSA tools have limitations to accommodate all the above facilities at once. Typically, utilities possess the NB data of their network in the Energy Management System (EMS) in CIM format. EMS platforms convert the NB model to the BB Model via Network Topology Processing (NTP) [15] which is further used by other analysis tools in EMS. However, these converted BB models are not accessible directly to external TSA tools. A compatible CIM file or custom data files in NB format must be manually created for NB models to be used in TSA tools. However, as they use Y-Bus/ power flow-based network solutions, they can only mimic the end effect of CB operations in a station due to relay decisions. Moreover, most test systems used in TSA by researchers today are available as BB models only. So, creating an NB model from a BB model in an automated manner is essential. In [18], we have proposed an algorithmic approach to convert BB models to NB models for different types of station configurations.

Developing a generalized transient stability program including conversion of BB model to NB model, populating all the relays automatically, associating the CBs with the relays and coordinating the timings for large systems to replicate the practical scenarios is a non-trivial task [15]. The following are the contributions of the paper to address the limitations of existing TSA tools.

- A unified framework based on the methods proposed in [17], [18] and discussion on the programmatic approaches to automatically populate relays (distance relays, differential relays and under/over frequency relays), CBs associated with each relay and timers of the relays and CBs in time domain simulation for TSA.
- Systematic approach for time coordination between the simulation time step, protective relays trip time in differ-

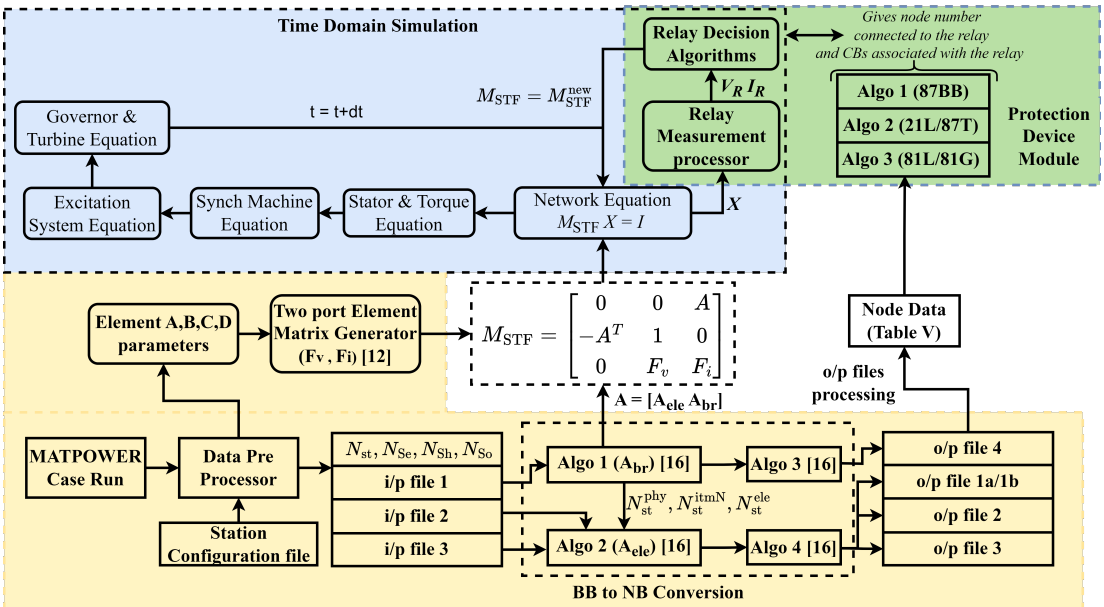


Fig. 1: Node Breaker Model Based Transient Stability Simulation including Protective Devices

ent zones and their associated CBs operating time.
- A comparison of the results and execution time using the proposed framework for WECC 9 bus system with the commercial software PSS®E.
- Simulation results showcasing the flexibility and scalability of the framework on the New England 39 bus system and the Polish 2383 bus system.

The proposed framework will help in new commercial TSA tools development which will alleviate the manual diligence required for creating practical cascade scenarios due to relay and CB operations in substations.

III. Proposed Simulation Framework

This section explains the proposed simulation framework, from data preparation to time domain simulation. Fig. 1 describes the framework in form of a block diagram. The proposed approach shown in Fig.1 builds the NB model of the system in the form of Sparse Tableau, which allows the modeling of CB operations that occur due to relay decisions realistically. In this paper, we use the methodology proposed in [18] for BB to NB model conversion, as shown in Fig.1. The test system data is taken from MATPOWER in BB format. A CIM interface can also be used here. After processing this BB data, input files containing series, shunt and source elements data, as shown in Table I - Table III, will be generated. Using these input files, Algorithms 1-4 in [18] create the output data files in NB format, as shown in Table IV.

For the time domain simulations, the network equations of the NB model are represented in the form of Sparse Tableau (M_{STF}). In this approach, each element is considered a linear two port element and can be represented by its ABCD parameters as follows

$$\overbrace{\begin{bmatrix}1 & -A_{\mathrm{l}}\\ 0 & -C_{\mathrm{l}}\end{bmatrix}}^{F_{\mathrm{v}}}\begin{bmatrix}v_{\mathrm{a}}\\ v_{\mathrm{b}}\end{bmatrix}+\overbrace{\begin{bmatrix}0 & B_{\mathrm{l}}\\ 1 & D_{\mathrm{l}}\end{bmatrix}}^{F_{\mathrm{i}}}\begin{bmatrix}i_{\mathrm{a}}\\ i_{\mathrm{b}}\end{bmatrix}=0 \tag{1}$$

CBs as a two port element can be represented as

$$\overbrace{\begin{bmatrix}\tau & -\tau\\ 0 & 0\end{bmatrix}}^{F_{\mathrm{v}}}\begin{bmatrix}v_{\mathrm{a}}\\ v_{\mathrm{b}}\end{bmatrix}+\overbrace{\begin{bmatrix}1-\tau & \tau\times CB_{\mathrm{res}}\\ \tau & 1\end{bmatrix}}^{F_{\mathrm{i}}}\begin{bmatrix}i_{\mathrm{a}}\\ i_{\mathrm{b}}\end{bmatrix}=0 \tag{2}$$

where, $\tau = 1$ and $\tau = 0$, represent the closed and open positions of circuit breaker respectively and CB_{res} is the CB resistance (we use 1e-6 pu). These element level F_{v} and F_{i} are constituted together to form system level F_{V} and F_{I}. Formation of F_{V} and F_{I} are given in [17]. Matrix A is the incidence matrix, which can be further divided into A_{br} and A_{ele}, where A_{br} is the $Node\ to\ CB$ incidence matrix and A_{ele} is the $Node\ to\ Element$ incidence matrix. *Algorithm 1* and *Algorithm 2* in [18] give A_{br} and A_{ele}. The system is represented via KVL and KCL equations. Hence, all the system equations can be constituted in the form of :

$$M_{\mathrm{STF}}=\begin{bmatrix}0 & 0 & A\\ -A^{T} & 1 & 0\\ 0 & F_{\mathrm{V}} & F_{\mathrm{I}}\end{bmatrix};\mathbf{X}=\begin{bmatrix}V\\ v\\ i\end{bmatrix};\mathbf{I}=\begin{bmatrix}I\\ 0\\ 0\end{bmatrix} \tag{3}$$

In transient stability simulation, $M_{\mathrm{STF}}X = I$ is solved to give X. Here, I is the current injection vector, and X is a vector containing node voltages and voltages and currents at each port of elements. The resultant vector X is sent to the Protective Devices module. This module's structure and functioning are described in detail in the next section. The relay algorithm processes the measurements and trips CBs, if any. The change in relay status is reflected via changing the corresponding F_{v} and F_{i} entries as shown in (2) in M_{STF} in the next time loop.

TABLE I: Input file 1 format

MATPOWER Bus Number	Configuration	$N_{\mathrm{st}}^{\mathrm{Se}}$	$N_{\mathrm{st}}^{\mathrm{Sh}}$	$N_{\mathrm{st}}^{\mathrm{So}}$

TABLE II: Input file 2 format

MATPOWER Bus Number	Configuration	Indexes of Series Element connected	Indexes of Shunt Element connected	Indexes of Source Element connected

TABLE III: Input file 3 format

Series Element Index	MATPOWER Fom Bus	MATPOWER To Bus

TABLE IV: NB Data file format

File1a	$Line_{\mathrm{i}}$	From Node	From Station	To Node	To Station
File1b	$Trafo_{\mathrm{i}}$	From Node	From Station	To Node	To Station
File2	$Load_{\mathrm{i}}$	Node	Station		
File3	Gen_{i}	Node	Station		
File4	CB_{N}	From Node	To Node	Station	

TABLE V: Node Data

Node Number(n)	Total Elements Connected ($n.N_{\mathrm{node}}^{\mathrm{ele}}$)	$n.eleType(1)$	...	$n.eleType(N_{\mathrm{node}}^{\mathrm{ele}})$
		$n.eleNo(1)$	...	$n.eleNo(N_{\mathrm{node}}^{\mathrm{ele}})$

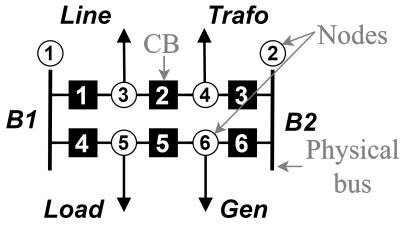


Fig. 2: Substation in Node-Breaker model with Breaker and Half configuration

A. Automatic Placement of Relays in Network

An NB model for a substation with Breaker and Half (BAH) configuration is shown in Fig. 2. The final output of the algorithms in [18] gives NB data as in Table. IV. $File\,1a$ and $File\,1b$ contains information about all the lines and transformers (series elements), $File\,2$ contains information about loads (shunt elements), and $File\,3$ contains information about the generators (source elements). The connecting nodes of all the CBs in the network are stored in $File\,4$. After the NB model of the network is developed, the NB data is presented

Algorithm 1: Busbar differential relays ($R_{87\mathrm{BB}}$)

1: Assume: $N_{\mathrm{cum}}^{\mathrm{nodes}}(0) = 0$;
2: $r_{\mathrm{num}} \leftarrow 1$; // index for relay number
3: **for** i = 1:N_{st} **do**
4: **for** j = 1:$N_{\mathrm{st}}^{\mathrm{phy}}$ **do**
5: $CB_{\mathrm{num}} \leftarrow 0$ // index that stores associated CB number to relay
6: $R_{87\mathrm{BB}}(r_{\mathrm{num}}).Node \leftarrow (N_{\mathrm{cum}}^{\mathrm{nodes}}(i-1)+j)$ // physical buses in a station are numbered first
7: **for** k = 1:$R_{87\mathrm{BB}}(r_{\mathrm{num}}).Node.N_{\mathrm{node}}^{\mathrm{ele}}$ **do**
8: **if** $R_{87\mathrm{BB}}(r_{\mathrm{num}}).Node.eleType(k) == CB$ **then**
9: $CB_{\mathrm{num}} \leftarrow CB_{\mathrm{num}}$+1
10: $R_{87\mathrm{BB}}(r_{\mathrm{num}}).CB(CB_{\mathrm{num}}) \leftarrow R_{87\mathrm{BB}}(r_{\mathrm{num}}).Node.eleNo(k)$ // CB number is associated to relay
11: **end if**
12: **end for**
13: $R_{87\mathrm{BB}}(r_{\mathrm{num}}).CB_{\mathrm{total}} \leftarrow CB_{\mathrm{num}}$
14: $r_{\mathrm{num}} \leftarrow r_{\mathrm{num}} + 1$
15: **end for**
16: **end for**

Algorithm 2: Line Distance Relays ($R_{21\mathrm{L}}$)/ Transformer differential relays ($R_{87\mathrm{T}}$)

1: $r_{\mathrm{num}} \leftarrow 1$
For $R_{87\mathrm{T}}$, replace *line*→*trafo* & $R_{21\mathrm{L}}$→$R_{87\mathrm{T}}$
2: **for** *line*=1:N^{line} **do**
3: $R_{21\mathrm{L}}(r_{\mathrm{num}}).Node \leftarrow line.FromNode$
4: $R_{87\mathrm{T}}(r_{\mathrm{num}}).Node1 \leftarrow trafo.ToNode$ // skip for $R_{21\mathrm{L}}$
5: $R_{21\mathrm{L}}(r_{\mathrm{num}}+1).Node \leftarrow line.ToNode$ // skip for $R_{87\mathrm{T}}$
6: $CB_{\mathrm{num}} \leftarrow 0$
7: **for** k = 1:$line.FromNode.N_{\mathrm{node}}^{\mathrm{ele}}$ **do**
8: **if** $line.FromNode.eleType(k) == CB$ **then**
9: $CB_{\mathrm{num}} \leftarrow CB_{\mathrm{num}}$+1
10: $R_{21\mathrm{L}}(r_{\mathrm{num}}).CB(CB_{\mathrm{num}}) \leftarrow line.FromNode.eleNo(k)$
11: **end if**
12: **end for**
13: $R_{21\mathrm{L}}(r_{\mathrm{num}}).CB_{\mathrm{total}} \leftarrow CB_{\mathrm{num}}$ // skip for $R_{87\mathrm{T}}$
14: $r_{\mathrm{num}} \leftarrow r_{\mathrm{num}} + 1$ // skip for $R_{87\mathrm{T}}$
15: $CB_{\mathrm{num}} \leftarrow 0$ // skip for $R_{87\mathrm{T}}$
16: **for** k = 1:$line.ToNode.N_{\mathrm{node}}^{\mathrm{ele}}$ **do**
17: **if** $line.ToNode.eleType(k) == CB$ **then**
18: $CB_{\mathrm{num}} \leftarrow CB_{\mathrm{num}}$+1
19: $R_{21\mathrm{L}}(r_{\mathrm{num}}).CB(CB_{\mathrm{num}}) \leftarrow line.ToNode.eleNo(k)$ // consecutive relay for pilot protection only for $R_{21\mathrm{L}}$
20: **end if**
21: **end for**
22: $R_{21\mathrm{L}}(r_{\mathrm{num}}).CB_{\mathrm{total}} \leftarrow CB_{\mathrm{num}}$
23: $r_{\mathrm{num}} \leftarrow r_{\mathrm{num}} + 1$
24: **end for**

in the format shown in Table IV. Using this data, relays can be placed for elements in network. All the data in File(1)-(4) is stored in arrays. The given data form a new vector of structures that stores node information as given in Table V. Here, n is the node number in the system, $n.N_{\mathrm{node}}^{\mathrm{ele}}$ gives the total number of elements (CB/line/trafo/load/generator) connected to node n. The members $n.eleType(k)$ and $n.eleNo(k)$ give element type and element number of the k^{th} element connected to the node.

For this paper we have considered differential relays for busbars, distance relays with Mho characteristics for lines, differential relays for transformers and under/over frequency relays for load/generator. Since these relays protect all the components, other relays are not considered. Although any other relay functions can be employed. Relay information is stored in arrays denoted via ANSI code for each type of relay. For example, $R_{87\mathrm{BB}}$, $R_{21\mathrm{L}}$, $R_{87\mathrm{T}}$, $R_{81\mathrm{L}}$, $R_{81\mathrm{G}}$ are

Algorithm 3: Load Frequency Relays ($R_{81\mathrm{L}}$)/ Generator Frequency Relays ($R_{81\mathrm{G}}$)

1: $r_{\mathrm{num}} \leftarrow 1$
For $R_{81\mathrm{G}}$, replace *load*→*gen* & $R_{81\mathrm{L}}$→$R_{81\mathrm{G}}$
2: **for** *load*=1:N^{load} **do**
3: $R_{81\mathrm{L}}(r_{\mathrm{num}}).Node \leftarrow load.Node$
4: $CB_{\mathrm{num}} \leftarrow 0$
5: **for** k = 1:$load.Node.N_{\mathrm{node}}^{\mathrm{ele}}$ **do**
6: **if** $load.Node.eleType == CB$ **then**
7: $CB_{\mathrm{num}} \leftarrow CB_{\mathrm{num}}$+1
8: $R_{81\mathrm{L}}(r_{\mathrm{num}}).CB(CB_{\mathrm{num}}) \leftarrow load.Node.eleNo(k)$
9: **end if**
10: **end for**
11: $R_{81\mathrm{L}}(r_{\mathrm{num}}).CB_{\mathrm{total}} \leftarrow CB_{\mathrm{num}}$
12: $r_{\mathrm{num}} \leftarrow r_{\mathrm{num}} + 1$
13: **end for**

the relay arrays that store information of busbar differential, distance, transformer differential, under and over frequency relays respectively. The length of an array R_{X} is N^{X} where X is the ANSI code. Each element of the array is a structure. For example, $R_{87\mathrm{BB}}(k)$ is the structure for k^{th} differential relay. $R_{87\mathrm{BB}}(k).Node$ stores the node number of busbar protected, $R_{87\mathrm{BB}}(k).CB_{\mathrm{total}}$ stores total number of CBs connected to the node and $R_{87\mathrm{BB}}(k).CB(i)$ ... $R_{87\mathrm{BB}}(k).CB(CB_{\mathrm{total}})$ stores CB numbers that are associated with the relay. The same convention is used for other relays $R_{21\mathrm{L}}$, $R_{81\mathrm{L}}$ and $R_{81\mathrm{G}}$. Algorithms are provided for the placement of relays in the system. Placement of relays means storing relay number, type, element it is protecting and the CBs associated with it. The key idea behind algorithms is to identify the node to which the element to be protected is connected from Table V. Each element connected to that node is spawned. If it is a CB, the CB number is stored as one of the associated CBs to the relay protecting the element. Then, same is repeated for next element with an incremented relay number. The algorithms provided in this paper currently populate 3 types of relays automatically in the following manner:

- Differential relays for busbars are populated bus number-wise using Algorithm-1.
- Distance relays for lines and differential relays for transformers are populated line/transformer number-wise using Algorithm-2.
- Frequency relays are populated load/generator number-wise using Algorithm-3.

Algorithm-1 shows the steps for placement of differential relays for busbar protection in the network. The first consecutive node numbers in each station are assigned to physical buses in the node data (Table V). Each busbar differential relay is associated to these physical buses in the Algorithm; step 6. All the CBs connected to the nodes are associated to the relays in step 10. Algorithm-2 shows the steps for placement of distance relays and differential relays for lines and transformers connected at each station in the network. Here, the relays associated with the two ends of the line are consecutively numbered; see steps 3 and 5. This gives flexibility in identifying the CBs that should be tripped for pilot protection. The CBs connected to the $FromNode$ and $ToNode$ are associated with the respective relays in steps 10 and 19 respectively. The same logic can be used to place

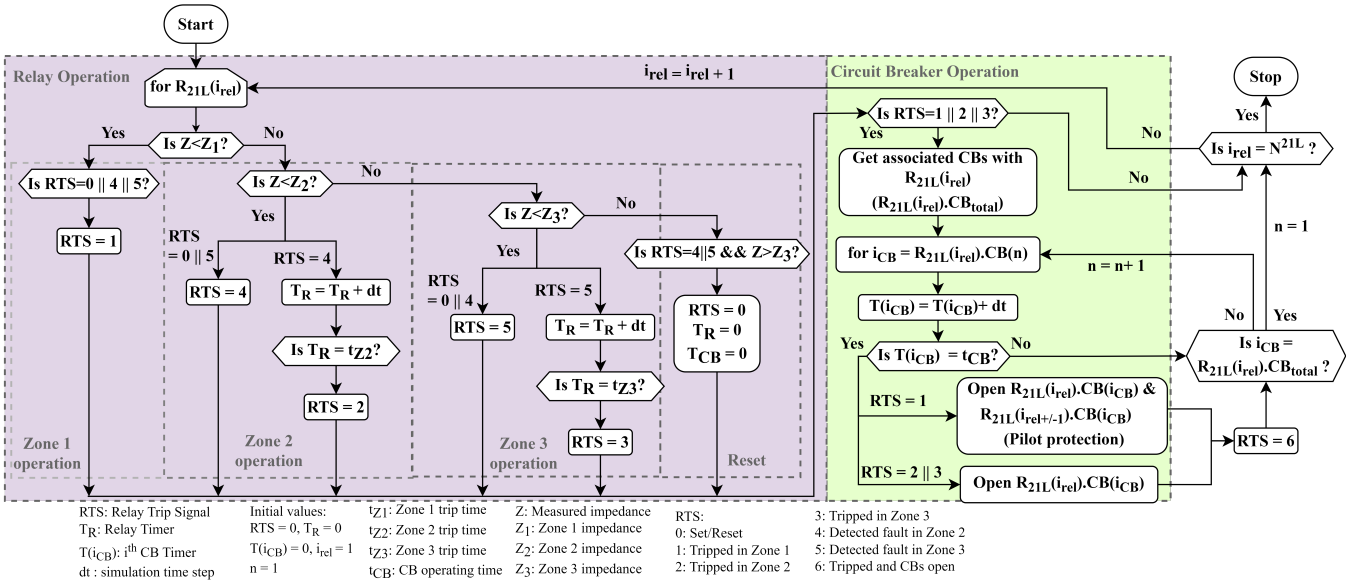


Fig. 3: Simulation Time Coordination between different Zones of Distance Relay and associated Circuit Breakers

differential relays for transformers by replacing the term *line* with *trafo* and associating both ends to a single relay. Step 4 can be performed instead of step 5 and steps 12,13 and 14 can be skipped. Algorithm-3 gives placement of frequency relays for all loads in the same way. The same algorithm is used for placing relays for generators. Any backup protection relays for any element can also be placed similarly. Once the relays are placed, the relay settings can be calculated based on differential characteristics [19] for $R_{87\mathrm{BB}}$, Mho characteristics and adjacent line impedances for $R_{21\mathrm{L}}$ [20] and frequency settings for $R_{81\mathrm{L/G}}$ [21].

B. Relay and Circuit Breaker Trip-Time Coordination

This section provides the framework for time coordination between relays and CBs in simulation. In simulation, the network solutions give voltages at each node and current at each port ($M_{\mathrm{STF}}\mathbf{X} = I$). These node voltages and current measurements are further processed to give two arrays V_{R} and I_{R} respectively. V_{R} stores voltage measurements for relays that require voltage as an input. I_{R} stores current measurements for relays that require current as an input. Since the identification number for the relays is already assigned via Algorithm 1-3, the relay measurements are arranged in the following form, so that it is easier to send arrays to the respective relay functions:

$$V_{\mathrm{R}} = \begin{bmatrix} V_{\mathrm{R21L}} \\ V_{\mathrm{R81L}} \\ V_{\mathrm{R81G}} \end{bmatrix}; I_{\mathrm{R}} = \begin{bmatrix} I_{\mathrm{R87BB}} \\ I_{\mathrm{R21L}} \\ I_{\mathrm{R81L}} \\ I_{\mathrm{R81G}} \end{bmatrix}$$

Each section of V_{R} and I_{R} is sent to respective relay algorithms sequentially (Fig. 1). For a particular time loop, V_{R} and I_{R} will be same for any relay, hence the sequence of relays does not matter. In implementation, we have considered sequence *Differential* $\rightarrow$ *Distance* $\rightarrow$ *Under/Overfrequency*. The differential and frequency relays are definite time relays. The distance relays have different zones of operation and trip times.

Fig. 3 shows time coordination between different zones of a distance relay and its associated CBs. The logic starts with the first distance relay and calculates measured impedance Z from corresponding entrance in V_{R21} and I_{R21}. First, it checks if Z comes inside $Zone\,1$ impedance circle. If true, it trips the relay instantaneously updating Relay Trip Signal (RTS) to 1 and starts the CB timer. If false, it checks if Z comes inside $Zone\,2$ impedance circle. If it comes in this zone and the relay has already detected fault in $Zone\,3$ (RTS=5) or not detected at all (RTS=0) in previous time step, the relay updates RTS to 4 i.e. in $Zone\,2$. If the relay has already detected fault in $Zone\,2$, it increments the relay timer T_{R} with one time step and checks if it is equal to $Zone\,2$ timer t_{Z2}. If $T_{\mathrm{R}} = t_{\mathrm{Z2}}$, it trips the relay in $Zone\,2$, updates RTS=2 and starts the CB timer. If Z does not come inside $Zone\,2$ as well, it is checked for $Zone\,3$. If the relay has already detected a $Zone\,2$ fault or nothing in previous time loop then RTS is updated to 5, i.e. detects fault in $Zone\,3$. If the relay has already detected a $Zone\,3$ fault then it simply increments T_{R} with one time step. If T_{R} reaches $Zone\,3$ timer then RTS is updated to 3, i.e. relay is tripped and the CB timer starts. If Z is out of all the impedance circles at any time loop, the CB timer, relay timer and RTS are set to 0. This part of the logic is depicted in the left portion of Fig.3 i.e. *Relay Operation*. This part is identical for both the Y-Bus and Sparse Tableau based approaches. In Y-Bus approach each relay is considered to have only two hypothetical CBs with a common timer to open/close at both ends. An average delay is added to account for the CB opening time before modifying the Y-Bus. Timing coordination program is trivial in Y-Bus approach.

In the proposed approach we explicitly model the CB operations and detailed station configurations. The CB operation is achieved in the network matrix (M_{STF}) by changing the

TABLE VI: Relay Models considered in PSS®E

Relay	Characteristic	Attributes
Distance	Mho	Zone 1/2/3 pick-up time, Zone 1/2/3 reach, Zone 1/2/3 center line angle, Zone 1/2/3 centre distance, self/transfer trip breaker time, self/transfer trip reclosure time
Load	Under-frequency load shed	first load shed point (Hz), first point pick up time, first fraction of load shed, Breaker time

corresponding F_{v} and F_{i} entries. The time coordination is essential between transient simulation time step, relay operating time and their associated CBs operation time. The right part of Fig. 3 i.e. *Circuit Breaker Operation* gives the flow chart for the proposed approach. In Fig.3, while the Relay Trip Signal (RTS) reaches 1/2/3 values, the next step is to open the associated CBs. The program must keep a vector of associated CBs ($R_{21\mathrm{L}}(i_{\mathrm{rel}}).CB(n)$) with each relay and the total number of CBs to be tripped ($R_{21\mathrm{L}}(i_{\mathrm{rel}}).CB_{\mathrm{total}}$) based on the station configuration. For a BAH scheme, a minimum of 2 CBs must be tripped for each relay, and a pilot trip command must be used to open 2 CBs at the other end of the station. As soon as the CB timer for associated i^{th} CB to relay i.e. $T(i_{\mathrm{CB}})$ starts to increment due to relay tripping in any of the zones, the relay condition $Z < Z_{1/2/3}$ does not matter. The CB statuses are updated to open CB as soon as the CB timer saturates ($T(i_{\mathrm{CB}}) = t_{\mathrm{CB}}$) and RTS is updated to 6. If tripped in $Zone\,1$, CBs are opened for both sides of the line (pilot protection) otherwise CBs are opened for one side of the line.

This process is repeated for all the distance relays in the system in a time loop incremented with the simulation time step. The *Circuit Breaker Operation* block will remain same for definite time relays with $R_{21\mathrm{L}}$ changed to the respective relay array. Meanwhile, the *Relay Operation* block will have a single zone of operation.

IV. Results

This section shows various scenarios that can be simulated using the proposed framework with relays and CBs in TSA of power system networks. WECC 3 generator 9 bus system is employed to validate the results of the proposed framework with commercial software PSS®E 34. After that, the proposed framework is applied to the New England 10 generator 39 bus system and the Polish 327 generator 2383 bus system. Fault conditions with the normal operation of CBs are compared with breaker failure operations after relay tripping. Since BAH is the most popular configuration used for transmission substations, each bus in the system is expanded to a BAH configuration. In simulations, we use a $Zone\,2$ timer of $0.26667s$ and $Zone\,3$ timer of $0.5s$ and $Zone\,1$ as instantaneous. We also used a fixed time delay of $0.05s$ (3 cycles) to account for the finite CB opening time after the trip command is issued.

For each case, the substation diagram for the concerned substations are provided with relay location assigned by algorithms discussed in Section III-A. In substation diagrams, differential relays are denoted by Df and distance relays are denoted by Ds. To keep the diagrams simple and understandable the relays that are not operating in fault case are not shown (transformer differential, under-over frequency for loads and generators). The neighbor substations are shown as buses. Bus fault and line fault are created and noteworthy relays and CB currents are plotted. In all the cases fault impedance Z_{f} is taken as 1e-4 pu.

Table VI shows the attributes for distance and load relays selected in PSS®E. The settings are calculated for these attributes based on [20] and relays are populated manually in PSS®E. Only one shed point at a frequency of 59 Hz is selected for load relays. Since PSS®E does not facilitate differential relays, the differential relay operation is mimicked in PSS®E by self clearing fault. PSS®E does not calculate settings instead it takes them as input. Hence, the settings are calculated via a separate program and given as input to both PSS®E and the proposed approach (PA).

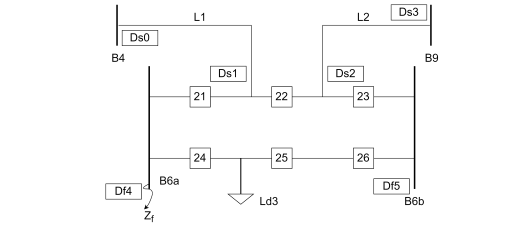
Table VII shows the total number of buses, nodes, CBs, relays and the associated timers to be coordinated for the test systems considered. In the Y-Bus based approach since there are no CBs associated with the model, the number of timers will be the same as the number of relays. It can be observed from the Table that, due to an increase in the intermediate nodes, physical buses, and CB elements the number of timers to be coordinated also increases with system size significantly in NB models. The proposed algorithms automatically populate, maintain, and update protective relays, CBs, and timers in the transient stability simulations.

A. WECC 9 bus system: Bus Fault (Fig. 4)

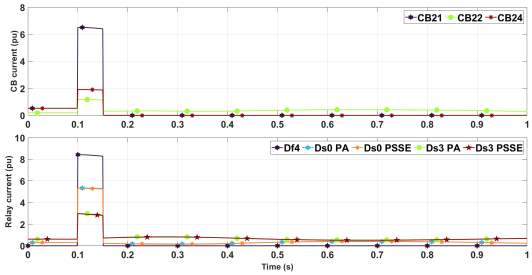
First, we discuss fault creation with PA. A 3ϕ balanced fault is created at bus $B6a$ at $0.1s$. The relay protecting $B6a$ is $Df4$ and the associated CBs are $CB21, CB24$. $Df4$ trips instantaneously. $Ds0$, $Ds3$ detect $Zone\,2$ fault. At $0.15s$, fault is cleared by opening the CBs $CB21, CB24$. Fig. 4b (top) shows the currents for $CB21, CB22, CB24$ went up during fault duration. After $CB21, CB24$ are open, the current through them becomes 0, and the current through $CB22$ comes to a post-fault value. Now, to create the same conditions in PSS®E, a bus fault is created at bus 6 with same

TABLE VII: Node-Breaker System Information

System	Buses	Nodes	CBs	CB Timers	Distance Relays	Distance Relay Timers	Differential Relays	U/OFLS Relays	U/OFLS Relay Timers
WECC 9Bus	18	48	45	45	18	18	18	6	6
New England	78	218	210	210	68	68	78	29	29
Polish	4766	14302	14304	14304	5452	5452	4766	2245	2245



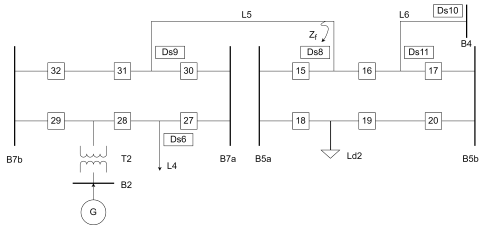
(a)



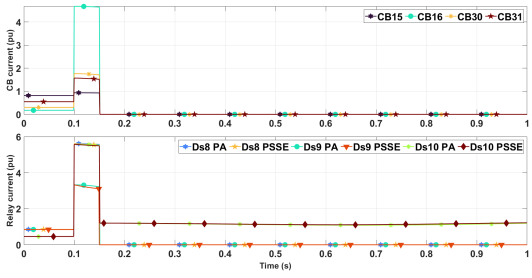
(b)

Fig. 4: WECC 9 bus system - Bus Fault at Bus 6a (a) Substation Configuration (b) CB and relay current measurements

fault impedance at $0.1s$ and removed at $0.15s$. Fig. 4b (bottom) compares currents seen by relays $Ds0$, $Ds3$ in PSS®E to those with the PA. It is observed that the relay currents match to a high extent.



(a)

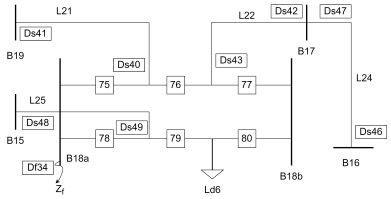


(b)

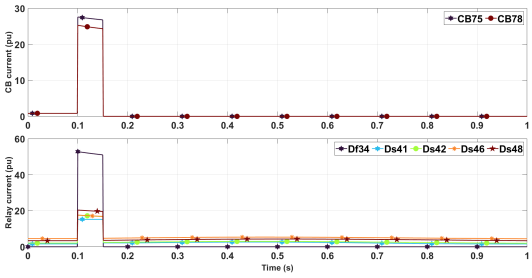
Fig. 5: WECC 9 bus system - Line fault at line 5 near station 5 (a) Station Configuration (b) CB current and relay measurements

B. WECC 9 bus system: Line Fault (Fig. 5)

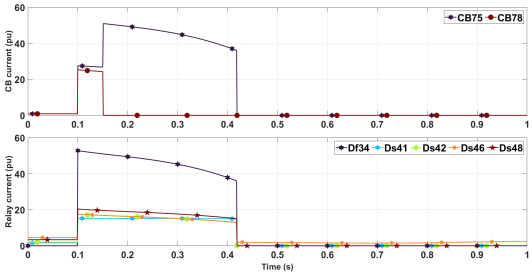
A 3ϕ balanced fault is created at one end of $Line$ 5 near station 5 at $0.1s$. $Ds8$ picks up in $Zone\,1$ and $Ds9$, $Ds10$ are picked up in $Zone\,2$. $Ds8$ trips instantaneously and sends a pilot trip command to $Ds9$ opening $CB15$, $CB16$, $CB30$, $CB31$ at $0.15s$. The currents for CBs can be seen from Fig. 5 (top). As soon as CBs are opened and $Line$ 5 is tripped, $Ds10$ gets reset. Fig. 5 (bottom) shows relay currents for $Ds8$, $Ds9$, $Ds10$ with the PA as well as PSS®E. The current seen by $Ds8$, $Ds9$ becomes 0 while $Ds10$ comes to a post fault value at $0.151s$. The similarity in waveforms of PA and PSS®E from the figures confirms the accuracy of PA. The time required for $10s$ of simulation with PSS®E is $\sim 0.068s$ and that with PA is $\sim 0.27s$. The runtime of PSS®E is less as it solves the network via Y-Bus not considering CBs whereas PA solves it with Sparse Tableau considering all the CBs.



(a)



(b)



(c)

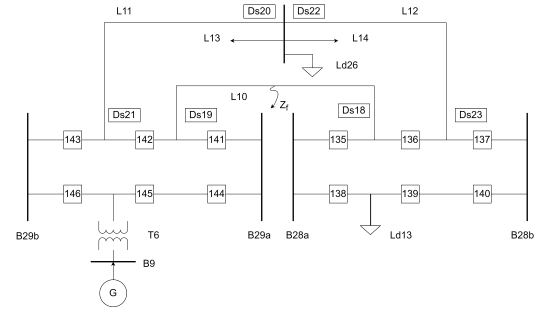
Fig. 6: New England system - Bus fault at bus 18a (a) Station Configuration (b) Normal operation (c) CB123 stuck

C. New England system: Bus Fault (Fig. 6)

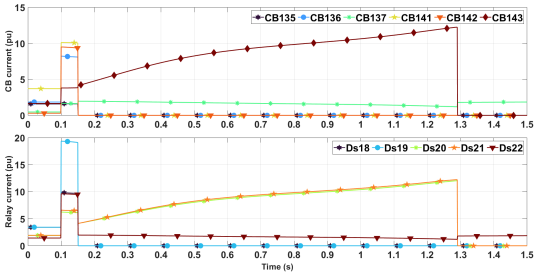
A 3ϕ balanced fault is created at bus $B18a$ at $0.1s$. The relay protecting $B18a$ is $Df34$ and the associated CBs are $CB75$, and $CB78$. $Df34$ trips instantaneously. $Ds41$, $Ds42$, $Ds48$ detect $Zone\,2$ fault and $Ds46$ detects $Zone\,3$ fault. At $0.15s$ the CBs associated to $Df34$ are opened and $B18a$ is isolated. $Ds41$, $Ds42$, $Ds46$ and $Ds48$ reset at $0.151s$. The system returns back to original. This scenario is called bus splitting. The CB and relay current waveforms are shown in Fig. 6b.

With same fault conditions, one of the CBs, say, $CB75$ gets stuck while operating. In that case, $Ds41$, $Ds42$, $Ds48$ trip in $Zone\,2$ at $0.368s$; The associated CBs in $Station$ 18 i.e. $CB76$, $CB77$, $CB78$ are opened at $0.481s$ tripping $L21$, $L22$, $L25$. As soon as $Line\,22$ is tripped $Ds46$ gets

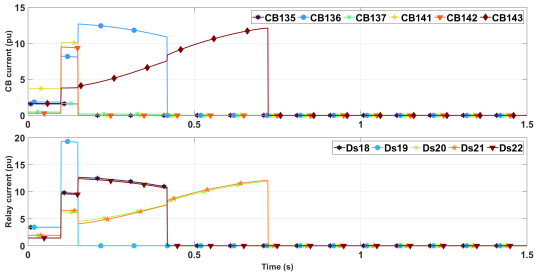
reset. The stuck breaker caused $N-3$ contingency (excluding $Ld6$).



(a)



(b)



(c)

Fig. 7: New England system - Line fault at Line 10 (a) Station Configuration (b) Normal operation (c) CB136 stuck

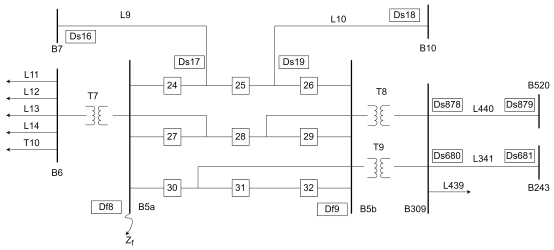
D. New England system: Line Fault (Fig. 7)

A 3ϕ balanced mid-line fault is created at $Line\,10$ at $0.1s$. $Ds18$ and $Ds19$ trip instantaneously in $Zone\,1$ at $0.1s$. $Ds20$ and $Ds22$ detect $Zone\,2$ faults. The CBs associated with $Ds18$ i.e. $CB135$, $CB136$ and $Ds19$ i.e. $CB141$, $CB142$ open at $0.15s$. This isolates $Line\,10$ from the system and the power flowing through $Line\,10$ now transfers to $Line\,11$. So generator current at $B9$ flows through $CB143$ to $Line\,11$. At $1.167s$, $Ds21$ detects $Zone\,2$ fault and trips at $1.24s$. $CB143$ opens at $1.29s$ isolating $Line$ 11. This isolates $Station$ 29 from the system.

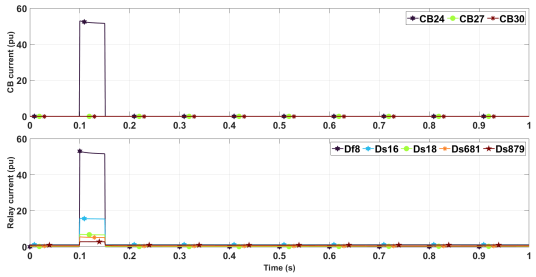
Another case is considered where $CB136$ gets stuck while operating. This makes $Ds22$ trip in $Zone\,2$ followed by tripping of $Ds21$ isolating $Station$ 29. But in this case the isolation happened much earlier than the normal operation with an increased contingency. The waveforms and timings can be seen from Fig. 7c.

E. Polish system: Bus Fault (Fig. 8)

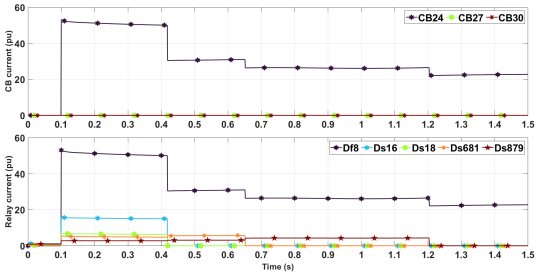
A 3ϕ balanced fault is created at bus $B5a$ at $0.1s$. The relay protecting $B5a$ is $Df8$ and the associated CBs are $CB24$, $CB27$, $CB30$. $Df8$ trips instantaneously. $Ds16$, $Ds18$ detect $Zone\,2$ fault and $Ds681$ detects $Zone\,3$ fault. At $0.15s$ the CBs associated to $Df8$ are opened and $B5a$ is isolated. $Ds16$, $Ds18$ and $Ds681$ reset at $0.151s$. Fig. 8b shows the current waveform of CBs and relays for the scenario. Fig. 8c shows the waveforms when $CB24$ gets stuck while operating. $Ds16$, $Ds18$ trip in $Zone\,2$ at $0.368s$ and open $CB25$, $CB26$ at $0.418s$. Hence, the current through $CB24$ gets reduced at $0.418s$ but the fault is still being fed via $T8$, $T9$. $Ds681$ trips at $0.601s$ in $Zone\,3$, tripping $Line$ 341 at $0.651s$, further reducing current through $CB24$. $Ds879$ trips in $Zone\,3$ at $1.153s$, tripping $Line$ 440 at $1.203s$. Note that only $CB27$, $CB30$ are opened in $Station$ 5 so the fault is still getting fed via $CB24$ from $B6$.



(a)



(b)



(c)

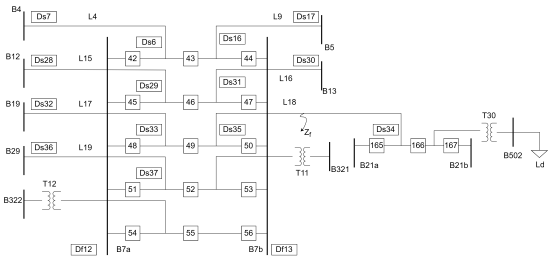
Fig. 8: Polish system - Bus fault at bus 5a (a) Station Configuration (b) Normal operation (c) CB24 stuck

F. Polish system: Line Fault (Fig. 9)

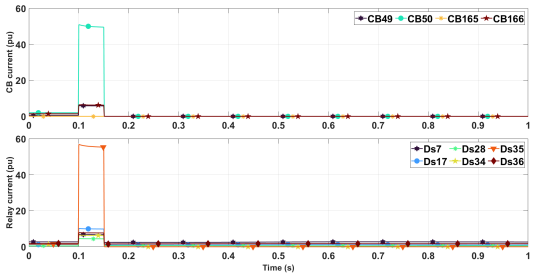
A 3ϕ balanced mid-line fault is created at $Line\,18$ at $0.1s$. $Ds34$, $Ds35$ detect faults in $Zone\,1$ and trip instantaneously.$Ds7$, $Ds17$, $Ds28$, $Ds30$, $Ds32$, $Ds36$ detect $Zone\,3$ faults. The CBs associated with $Ds34$, $Ds35$, i.e. $CB49$, $CB50$ and $CB165$, $CB166$ open at $0.15s$,

resetting all the other relays. Fig. 9b shows waveforms for the case.

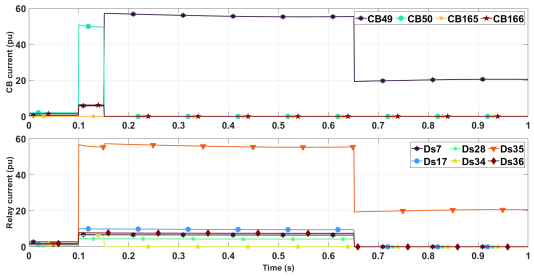
Fig. 9c shows currents for a case where $CB49$ gets stuck. All the other relays detecting fault in $Zone\,3$ trip at $0.601s$, opening associated CBs in their station at $0.651s$. While all the lines connected to $Station\ 7$ are tripped, $T12$ still connects $B7a$ to $B322$. This is still feeding current to fault via $CB49$.



(a)



(b)



(c)

Fig. 9: Polish system - Line fault at Line 18 (a) Station Configuration (b) Normal operation (c) CB49 stuck

From the above result cases it is evident that incorporating protective relays and CBs in TSA plays an important role in TSA. The CBs, when mal-operate, can result in a higher level of contingency than that in normal operations. It can also result in sustained faults in system which may lead to cascade events. With proposed framework, it is very convenient to represent the system in NB model with all the CBs, automatically place relays for each element and get associated CBs and trip any element of the system by opening the CBs. This can further help to perform substation level analyses and cascading analyses.

The simulation was carried out with a *cpp* code on Intel® Xeon® Silver 4110 CPU @ 2.10GHz 32GB RAM. Table VIII shows the break-up of time required for various functions for each time loop. The symbolic and numeric factorization of M_{STF} takes place before the time loop starts or when M_{STF} needs to be refactorized. The other functions in Table VIII are called in every time loop hence, contribute to the total simulation time. The majority of time is taken for backward/forward substitution to solve $M_{\text{STF}}X = I$. This is due to increased size of Sparse Tableau i.e. M_{STF}. The computational enhancement of Sparse Tableau is discussed in [22]. However, this paper does not focus on this. The time taken for relay computations are also provided. The time taken for a $10s$ simulation of the Polish system using the NB model with relays is $98s$, whereas, for the BB model without any relays, it is $19.2s$ regarding the same code setup. Although the simulation time with proposed approach is higher, it provides extreme flexibility for the operators to analyze the system. Parallelization approaches can be adopted to reduce the timings of the simulation further.

G. Parallel Processing Capability of Proposed Framework

Multiple cores can be employed to break down the proposed framework methodology to make full use of modern supercomputers. Since the placement of relays is performed only once before simulation, parallelization of the same will not affect much. Still, it could be done by assigning one type of relay placement to one CPU. The significant time reduction can be achieved by reducing the time of each loop. The Protection Devices Module can be decoupled from the differential and algebraic equations for transient simulation. Input vector X can be used from the previous time step to process the Protection Device Module in parallel. The drawback with this approach would be a lag of one time step in Protection Module action. However, it can be neglected as the expected nature of the waveform for TSA will remain the same, and all the timers will still be synchronized. With this modification, the extra computation time required for relays can be removed entirely from the simulation. The simulation time for New England system can be reduced by half and for Polish system by 34%. Also, parallel methods can be explored to reduce the forward/backward substitution, as it is the most time consuming part of the time loop.

TABLE VIII: Simulation Time (in ms)

Function	39 Bus	2383 Bus
Symbolic factorization	0.4	71.15
Numeric factorization	0.6	36.157
Forward/Backward substitution	0.17208	6.531
Differential relays computation	0.035	0.32
Distance relays computation	0.1428	1.94
Frequency relays computation	0.00428	0.0595
Total time for relays	0.182	2.3195
Differential Equation solver	0.0102	0.11712
States update	0.0013	0.0102
Total time ($10s$)	$1.16s$	$98s$

V. Conclusion

This paper provides a simulation framework to include protective relays and CBs in transient stability assessment. The Sparse Tableau Approach is used to represent the system in Node-Breaker model. Algorithms are derived for automatic placement of differential, distance and under/over-frequency relays in the system and CBs associated to them. The timer logic implemented for coordinating different zone timers of distance relay and CB timer is provided. The proposed framework is validated by comparing WECC 9 bus system results with commercial software PSS®E 34. Implementation on the New England 39 Bus system and the Polish 2383 Bus system is performed and scalability of the approach is discussed. The substation diagrams resulting from the proposed algorithms show the ease of implementation. Bus faults and line faults were created to check relay and CB operations. Each fault case with normal operation of relays and CBs is compared with the breaker failure operation. From the results it can be observed that CB maloperation can result in higher contingency as well as sustained faults in the system which can not be visualized with the Bus-Branch model of systems.

References

[1] L. G. Perez, A. J. Flechsig, and V. Venkatasubramanian, "Modeling the protective system for power system dynamic analysis," *IEEE Transactions on Power Systems*, vol. 9, no. 4, pp. 1963–1973, 1994.

[2] N. A. Samaan, J. E. Dagle, Y. V. Makarov, R. R. Diao, M. R. Vallem, T. B. Nguyen, L. E. Miller, B. G. Vyakaranam, F. K. Tuffner, M. A. Pai, J. Conto, and S. W. Kang, "Modeling of protection in dynamic simulation using generic relay models and settings," *IEEE Power and Energy Society General Meeting*, vol. 2016-Novem, 2016.

[3] A. Gopalakrishnan, S. G. Aquiles-Pérez, D. M. MacGregor, D. B. Coleman, P. Mcguire, K. W. Jones, J. Senthil, J. W. Feltes, G. Pietrow, and A. Bose, "Simulating the smart electric power grid of the 21st century–bridging the gap between protection and planning," in *Georgia Tech Protective Relaying Conference 2014*, 2014.

[4] I. Dobson, A. Flueck, S. Aquiles-Perez, S. Abhyankar, and J. Qi, "Towards incorporating protection and uncertainty into cascading failure simulation and analysis," in *2018 IEEE International Conference on Probabilistic Methods Applied to Power Systems (PMAPS)*, 2018, pp. 1–5.

[5] M. Abdi-Khorsand and V. Vittal, "Modeling protection systems in time-domain simulations: A new method to detect mis-operating relays for unstable power swings," *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 2790–2798, 2017.

[6] M. Liu, M. A. A. Murad, J. Chen, and F. Milano, "Modeling of protective relays for transient stability analysis," in *2020 IEEE Power & Energy Society General Meeting (PESGM)*, 2020, pp. 1–5.

[7] O. Patino, C. C. Soto, A. P. Grilo-Pavani, R. A. Ramos, R. M. Furlaneto, and I. Kocar, "A practical method to represent distance protection relays in transient stability simulations of lines connecting inverter-based resources," in *2023 IEEE Power & Energy Society General Meeting (PESGM)*, 2023, pp. 1–5.

[8] M. Abdi-Khorsand and V. Vittal, "Identification of Critical Protection Functions for Transient Stability Studies," *IEEE Transactions on Power Systems*, vol. 33, no. 3, pp. 2940–2948, 2018.

[9] R. Vakili, M. Khorsand, V. Vittal, B. Robertson, and P. Augustin, "An algorithmic approach for identifying critical distance relays for transient stability studies," *IEEE Open Access Journal of Power and Energy*, vol. 8, pp. 107–117, 2021.

[10] "Node-breaker modeling representation," *North American Electric Reliability Corporation (NERC), Tech. Rep., 2016.*

[11] G. Cardoso, J. G. Rolim, and H. H. Zurn, "Identifying the primary fault section after contingencies in bulk power systems," *IEEE Transactions on Power Delivery*, vol. 23, no. 3, pp. 1335–1342, 2008.

[12] K. Jiang and C. Singh, "New models and concepts for power system reliability evaluation including protection system failures," *IEEE Transactions on Power Systems*, vol. 26, no. 4, pp. 1845–1855, 2011.

[13] S. Robak, J. Machowski, M. Skwarski, and M. Januszewski, "Improvement of Power System Transient Stability in the Event of Multi-Phase Faults and Circuit Breaker Failures," *IEEE Transactions on Power Systems*, vol. 35, no. 3, pp. 2422–2430, 2020.

[14] K. R. Timalsena, R. Karki, and S. Bhattarai, "Methodology to incorporate stuck condition of the circuit breaker in reliability evaluation of electrical distribution networks," *IET Generation, Transmission and Distribution*, vol. 14, no. 21, pp. 4956–4964, 2020.

[15] R. Ramanathan, A. Popat, M. Papic, and O. Ciniglio, "Idaho power experience of implementing cascade analysis study using the node/breaker model," in *2017 IEEE Power & Energy Society General Meeting*, 2017, pp. 1–5.

[16] B. Park, J. Holzer, and C. L. DeMarco, "A sparse tableau formulation for node-breaker representations in security-constrained optimal power flow," *IEEE Transactions on Power Systems*, vol. 34, no. 1, pp. 637–647, 2019.

[17] V. Sahu and G. Gurrala, "Node breaker model based transient stability simulations using sparse tableau approach," in *2022 International Conference on Electrical, Computer and Energy Technologies (ICECET)*, 2022, pp. 1–6.

[18] ——, "Algorithm to convert power system network data from bus-branch model to node-breaker model," in *2023 IEEE Power and Energy Society General Meeting (PESGM)*, 2023, pp. 1–5.

[19] "Ieee guide for application of digital line current differential relays using digital communication," *IEEE Std C37.243-2015*, pp. 1–72, 2015.

[20] "Ieee guide for protective relay applications to transmission lines," *IEEE Std C37.113-2015 (Revision of IEEE Std C37.113-1999)*, pp. 1–141, 2016.

[21] "Ieee/iec international standard - measuring relays and protection equipment - part 118-1: Synchrophasor for power systems - measurements," *IEC/IEEE 60255-118-1:2018*, pp. 1–78, 2018.

[22] B. Park, "Computational enhancement of sparse tableau via block lu factorization for power flow studies," *IEEE Transactions on Power Systems*, vol. 38, no. 5, pp. 4974–4977, 2023.